

REMARKS

Claims 1-37 are currently pending in the subject application and are presently under consideration. The Examiner is thanked for courtesies extended during a discussion conducted on June 4, 2008. The main focus of the discussion was on deficiencies of the 35 U.S.C. §102(b) rejection. While the presented matter generally related to all the claims, the crux was upon claims 1, 22, 31, 33, and 34. Primarily, distinction between the claims and Ravindran reference was discussed. No agreements were reached. The discussion was conducted with Ronald Krosky (Reg. No. 58,564) and Examiner Chou. Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 1-37 Under 35 U.S.C. §102(b)

Claims 1-37 stand rejected under 35 U.S.C. §102(b) as being anticipated by Ravindran, Engineering Dynamic Real-Time Distributed Systems: Architecture, System Description Language, and Middleware, January, 2002. Ravindran does not teach each and every element of the claimed subject matter as recited in the subject claims.

A single prior art reference anticipates a patent claim only if it expressly or inherently describes *each and every limitation* set forth in the patent claim. *Trintec Industries, Inc. v. Top-U.S.A. Corp.*, 295 F.3d 1292, 63 USPQ2d 1597 (Fed. Cir. 2002); *See Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). *The identical invention must be shown in as complete detail as is contained in the ... claim.* *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). (emphasis added).

The claimed subject matter relates use of attribution to express management information in an application or service. Attribution is used to describe which part of the application or service code should be used to determine and/or correct health as well as to specify rules monitoring such aspects. In particular, independent claim 1 recites *a machine-implemented system that facilitates management of an application or service, comprising: an application or service for installation on the system; and an attribution component that facilitates attributing selected parts of code of the application or service with management information; where the system uses the management information to*

manage the installed application or service. Independent claims 22, 31, 33 and 34 recite similar limitations. While the examiner stated that the aforementioned limitation is anticipated by the cited reference (Ravindran), applicant's representative would like to respectfully explain in further detail the distinction between the cited reference and the subject claim including previously presented as well as new distinction. Ravindran does not teach or suggest the aforementioned novel aspects of applicants' claimed subject matter.

Ravindran provides for system and method for developing a dynamic real-time distributed system application in programming language. Operational requirement of the system such as timeliness and survivability are specified into a system description language. The dynamic model is used by resource management middleware strategies to perform resource management that achieves the timeliness and survivability requirement. The middleware techniques achieve the timeliness and survivability requirement through run-time monitoring and failure detection, diagnosis and dynamic resource allocation.

At page 2 and 3 of Final Office Action, it is erroneously asserted that Ravindran teaches *an attribution component that facilitates attributing selected parts of code of the application or service with management information*, with respect to independent claim 1. The reference (Ravindran) provides for a resource management architecture, wherein a system description language is used to specify the architectural-level description of the application such as composition and interconnections of application software and hardware resources and its timeliness and survivability requirements as desired quality of service (QoS). A static model of the system (an intermediate representation) is automatically constructed from the language specifications by a compiler of the language. The static intermediate representation is augmented with dynamically measured application performance characteristics by a language runtime system. The dynamic intermediate representation characterizes the state of the application and a middleware uses it for delivering the desired QoS to the application. *The application interfaces with the system description language runtime system and the middleware through application program interfaces* (See, Section 5). A language runtime system detects low quality of service (QoS) situations and performs diagnosis to determine causes of low QoS and notifies the middleware. The middleware analyzes the result of diagnosis to identify

possible recovery actions to improve the QoS. Typical actions taken by the middleware include replicating application programs of an overloaded task, migrating application programs from heavily loaded resources to less loaded resources and restarting an application program that failed (*See*, Section 6.1). Hence Ravindran provides for a system description language for specifying the architectural level description of the application, a language runtime system for dynamically measuring application performance characteristic and a middleware using dynamic performance characteristics for delivering the desired quality of service. However *the application interfaces with the system description language runtime system and the middleware through the application program interfaces*. More particularly, the resource management architecture including middleware and system description language runtime system is separate from the application code or separate piece of code from the application code and the application accesses and interfaces with the middleware and system description language runtime system through application program interfaces (APIs). However Ravindran does not contemplate an attribution component that facilitates attributing *selected parts of code of the application* with management information and the system using the management information to manage the installed application or service. Through this feature, the claimed subject matter facilitates a developer to avoid writing and maintaining two separate pieces of code—one for normal application processing and a second one to expose it to management. With attribution, the second body of code (like middleware and system description language run time system provided by Ravindran) and the associated maintenance and consistency issues are substantially eliminated making it much easier for developers to participate in management.

At page 8 of Office Action, it is erroneously asserted that Ravindran teaches *attributing selected parts of code of the application or service with management information and exposing the management information to a management system*, with respect to independent claim 22. The reference (Ravindran) provides for specification of an application program describing its name and attributes such as Boolean properties which indicate whether the application program can be replicated for survivability and whether the application program can be replicated for scalability to adapt the application program to increase in workloads (*See*, Section 7, System Description Language). Hence

Ravindran provides for describing attributes such as Boolean properties (yes or no) that indicate only whether the application program can be replicated for survivability and whether the application program can be replicated for scalability to adapt the application program to increase in workloads. Further Ravindran provides for describing attributes separately and including them in the specification and thereby increasing the code of application and the described attributes only indicates survivability and scalability of the application program. Nowhere Ravindran teaches or suggests *attributing selected parts of code of the application with management information and exposing the management information to a management system*.

At page 12 of the Final Office Action, it is erroneously asserted that Ravindran teaches that *the attributed parts of code are considered probes for use in determining health of the application*, with respect to dependent claim 37. The reference (Ravindran) provides for a middleware analyzing result of diagnosis, performed by a language runtime system, to identify possible recovery actions. Typical actions taken by the Middleware includes replicating application programs of an overloaded task, migrating application programs from heavily loaded resources to less loaded resources and restarting an application program that failed (*See*, Page 35, and Section 6.1). Hence Ravindran provides for only identifying possible recovery actions by the middleware for failed application programs which include replicating application programs of an overloaded task, migrating application programs from heavily loaded resources to less loaded resources and restarting an application program that failed. However the middleware uses a dynamic intermediate representation for identifying possible recovery actions and the dynamic intermediate representation which characterizes the state of the application, is created by a system description language runtime system *from whole of the application code* and not from selected parts of code of application attributed with management information. Further, the middleware *is a separate architecture than the application code* and the application program accesses and interfaces with the middleware through the application program interfaces (APIs). Hence Ravindran fails to teach or suggest *the attributed parts of code are considered probes for use in determining health of the application*.

Accordingly, applicants' representative respectfully submits that Ravindran fails

to teach or suggest all limitations of applicants' claimed subject matter as recited in amended independent claims 1, 22, 31, 33 and 34 (and claims that depend there from). Consequently, this rejection should be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP522US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,

AMIN, TUROCY & CALVIN, LLP

/Himanshu S. Amin/

Himanshu S. Amin

Reg. No. 40,894

AMIN, TUROCY & CALVIN, LLP
24TH Floor, National City Center
1900 E. 9TH Street
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731